BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

We claim:

1. A data cache for caching data in a computing device, the data cache operating at a plurality of levels in a memory hierarchy, the data cache comprising:

a plurality of subarrays for storing the data, each of the plurality of subarrays being divided into a plurality of individually addressable ways, each of the plurality of ways being individually configurable to operate at one of the plurality of levels in the memory hierarchy;

a decoder for addressing the data in the subarrays;

a global wordline connecting the decoder to each of the plurality of subarrays; and

a local wordline connecting the global wordline to each of the of plurality of ways in each of the plurality of subarrays.

2. The data cache of claim 1, wherein:

each of the plurality of subarrays has an end closest to the decoder; and

in each of the plurality of subarrays, the local wordline branches from the global wordline at the end closest to the decoder.

3. The data cache of claim 1, wherein the global wordline comprises a first plurality of repeater switches to isolate the plurality of subarrays from one another.

4. The data cache of claim 3, wherein, in each of the plurality of subarrays, the local wordline comprises a second plurality of repeater switches to isolate the plurality of ways from one another.

5. The data cache of claim 1, further comprising means for selectively disabling the plurality of subarrays.

6. The data cache of claim 5, further comprising means for selectively disabling the plurality of ways in each of the plurality of subarrays.

7. The data cache of claim 1, wherein the plurality of levels comprise L1 and L2.

8. The data cache of claim 1, wherein the plurality of levels comprise L2 and L3.

9. The data cache of claim 8, further comprising an L1 cache of fixed size.

10. A translation look-aside buffer for use in a computing device, the translation look-aside buffer comprising:

a plurality of increments, each for storing virtual page numbers and associated physical page numbers, each of the plurality of increments having an input and an output;

an input bus connected to the inputs of the plurality of increments;

an output bus connected to the outputs of the plurality of increments; and

switches in the input and output buses to disconnect the plurality of increments selectively from the input and output buses to vary a size of the translation look-aside buffer.

11. The translation look-aside buffer of claim 10, wherein the plurality of increments are connected in parallel to the input and output buses.

12. The translation look-aside buffer of claim 10, wherein each of the plurality of increments comprises:

a first memory for storing the virtual page numbers, the first memory having an input and an output; and

a second memory for storing the physical page numbers, the second memory having an input and an output;

and wherein:

the output of the first memory is connected to the input of the second memory;

the input of the first memory is the input of the increment; and

the output of the second memory is the output of the increment.

13. A method of reconfiguring a data cache for caching data in a computing device, the data cache operating at a plurality of levels in a memory hierarchy and comprising a portion having a variable size operating at a first level of the plurality of levels, the method comprising:

(a) storing performance information for the data cache;

(b) determining, from the performance information, whether the data cache has a miss rate exceeding a threshold;

(c) determining whether the variable size is equal to a maximum size; and

(d) if the miss rate exceeds the threshold and the variable size is not equal to the maximum size, controlling the data cache to increase the variable size.

14. The method of claim 13, further comprising:

(e) if the miss rate does not exceed the threshold or the variable size is equal to the maximum size, (i) determining, from the performance information, an optimal data cache configuration which optimizes a number of cycles per instruction in the computing device and (ii) setting the data cache to the optimal data cache configuration.

15. The method of claim 14, wherein, in each of a plurality of time periods during which the data cache operates, steps (a)-(c) and one of steps (d) and (e) are performed.

16. The method of claim 15, wherein each of the time periods is a fixed number of cycles of the computing device.

17. The method of claim 15, wherein each of the time periods is a time period in which the computing device performs a subroutine.

18. The method of claim 15, wherein:

the data cache is designated as either stable or unstable; and

steps (a)-(c) are performed only during intervals in which the data cache is designated as unstable.

19. The method of claim 18, further comprising, during intervals in which the data cache is designated as stable:

(f) determining, from the performance information, whether the data cache is actually unstable; and

(g) if the data cache is actually unstable, (i) designating the data cache as unstable and

(ii) setting the variable size to a minimum value.

20. The method of claim 19, wherein:

the performance indication comprises a hit counter for a second portion of the data cache which is outside the portion having the variable size; and

when the data cache is designated as stable and the hit counter is below a hit counter threshold, the second portion of the data cache is bypassed.

21. The method of claim 13, wherein:

the data cache comprises tag arrays and data arrays;

the first level is L1; and

in the portion having the variable size, the tag arrays and the data arrays are read in series.

22. A method of reconfiguring a translation look-aside buffer for use in a computing device, the translation look-aside buffer having a variable size, the method comprising:

(a) storing performance information for the translation look-aside buffer;

(b) determining, from the performance information, whether the translation look-aside buffer has a miss rate exceeding a first threshold;

(c) determining, from the performance information, whether the translation look-aside buffer has a usage less than a second threshold;

(d) if the miss rate exceeds the first threshold, controlling the translation look-aside buffer to increase the variable size; and

(e) if the use is less than the second threshold, controlling the translation look-aside buffer to decrease the variable size.

23. The method of claim 22, wherein, in each of a plurality of time periods during which the data cache operates, steps (a)-(c) and one of steps (d) and (e) are performed.

24. The method of claim 23, wherein each of the time periods is a fixed number of cycles of the computing device.